SensIT
Integration and Experimentation Update

BBN Technologies

Prakash Manghwani, Jeff Mazurek, Gail Mitchell, Ken Theriault

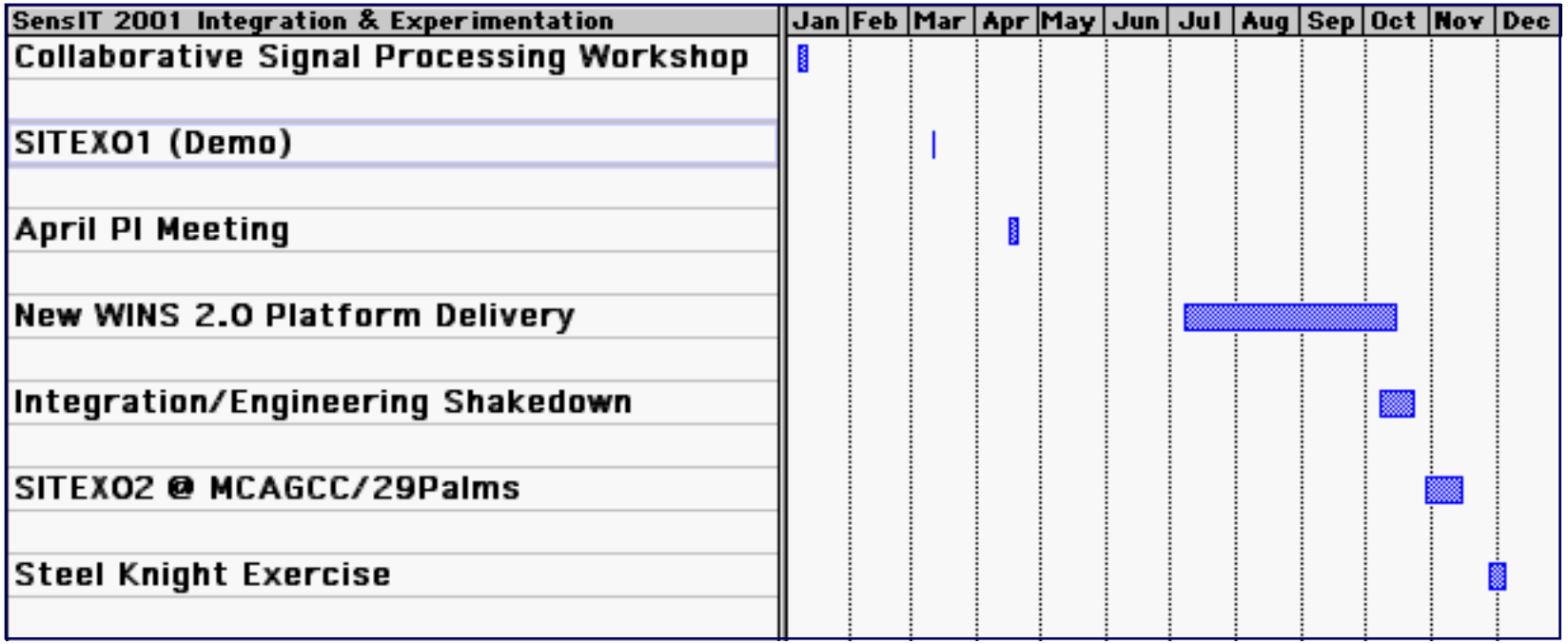
SensIT Goals/Challenge

- **Software technologies that can be applied to a variety of sensor nets, capable of carrying out multiple missions**
- **Integrated, robust S/W suite that enables ad-hoc sensor networks for diverse functions and scales**

- **<u>Integration</u>: “sensorware” - common software environment for experimentation in scheduling and controlling sensors, data, and communications.**
  - **combine the different technologies into a cohesive system**
  - **system level experimentation to test hypotheses and to investigate sensor network issues and tradeoffs.**
  - **Application level testing and experimentation**

- **Challenge Problem: Autonomous, real-time, tactical target localization, tracking, classification, and image**

2001 - Year in Review



- **Giant Steps toward common software environment**
  - **New Platform/OS (WINS NG 2.0 w/Linux)**
  - **Three Field Tests**
  - **SITEX02 Baseline S/W Definition**
  - **Baseline S/W Integration efforts (BBN Camb & SITEX02)**

SensIT Experiment - SITEX02 November 2001

Marine Corps Air Ground Combat Center Twentynine Palms, CA

Autonomous, distributed ground sensors that track moving vehicles and transmit processed information to a base camp display

800 m

600 m

400 m

200 m

Intersection

Total: 70 Nodes

Gateway/Imager

RF Ethernet

Base Camp

East Training Area
Prospect Intersection

SITEX02 Objectives

- Tactical localization & tracking with large distributed sensor networks: single target, multiple simultaneous targets
- Establish open, flexible, tactical sensor system and develop procedures for its use in further development and testing
  - Establish system performance baseline (e.g. network and software latencies)
- Install, and operate in the field, a large network of sensor nodes
  - Dual network: RF for tactical and IP for experiment support
  - Test WINS NG 2.0 nodes performance and reliability
- Collect time series data from a large set of nodes under controlled test conditions
- Developmental experimentation

Three Experiment Phases

1 - Set-up (Logistics)

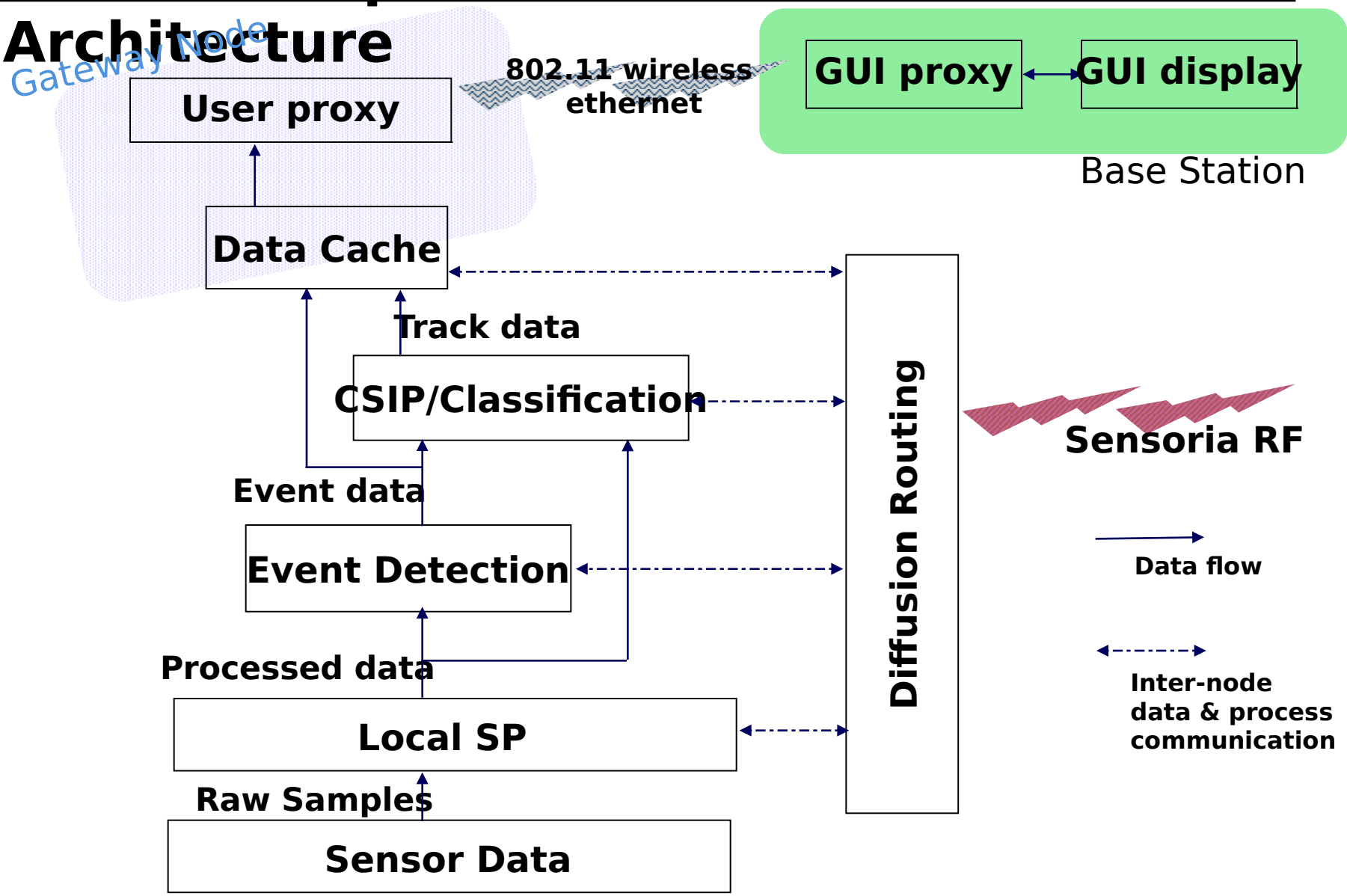
2 - Operational (Baseline) experiment

- Structured experiments using baseline H/W & S/W suite
- Exercise baseline system multi-target tracking/imaging capability
- Collect data to support further laboratory testing

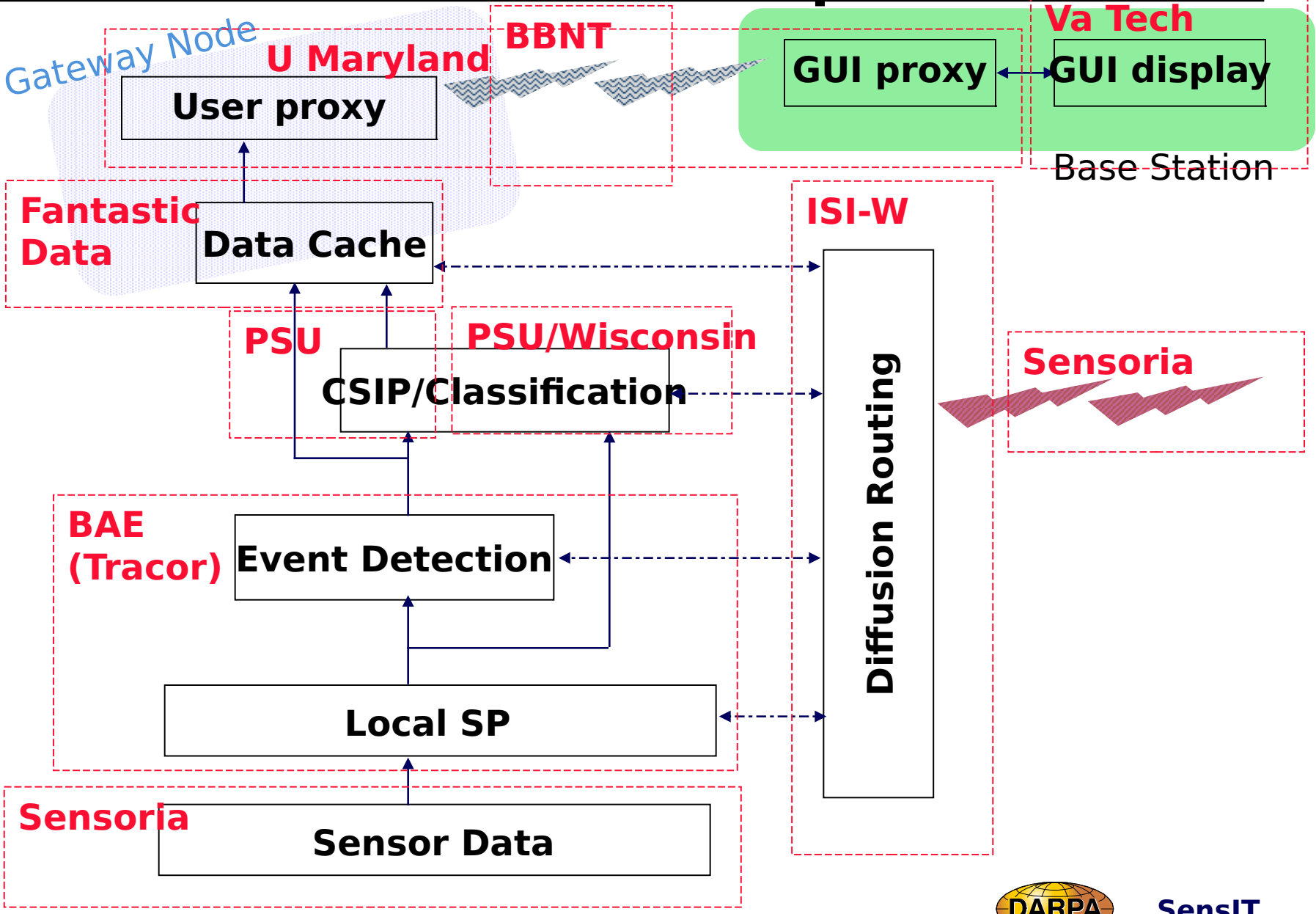
3 - Developmental experiments

- Real-time test and evaluation of developmental algorithms and processes
- Available for PI's to evaluate H/W & S/W performance

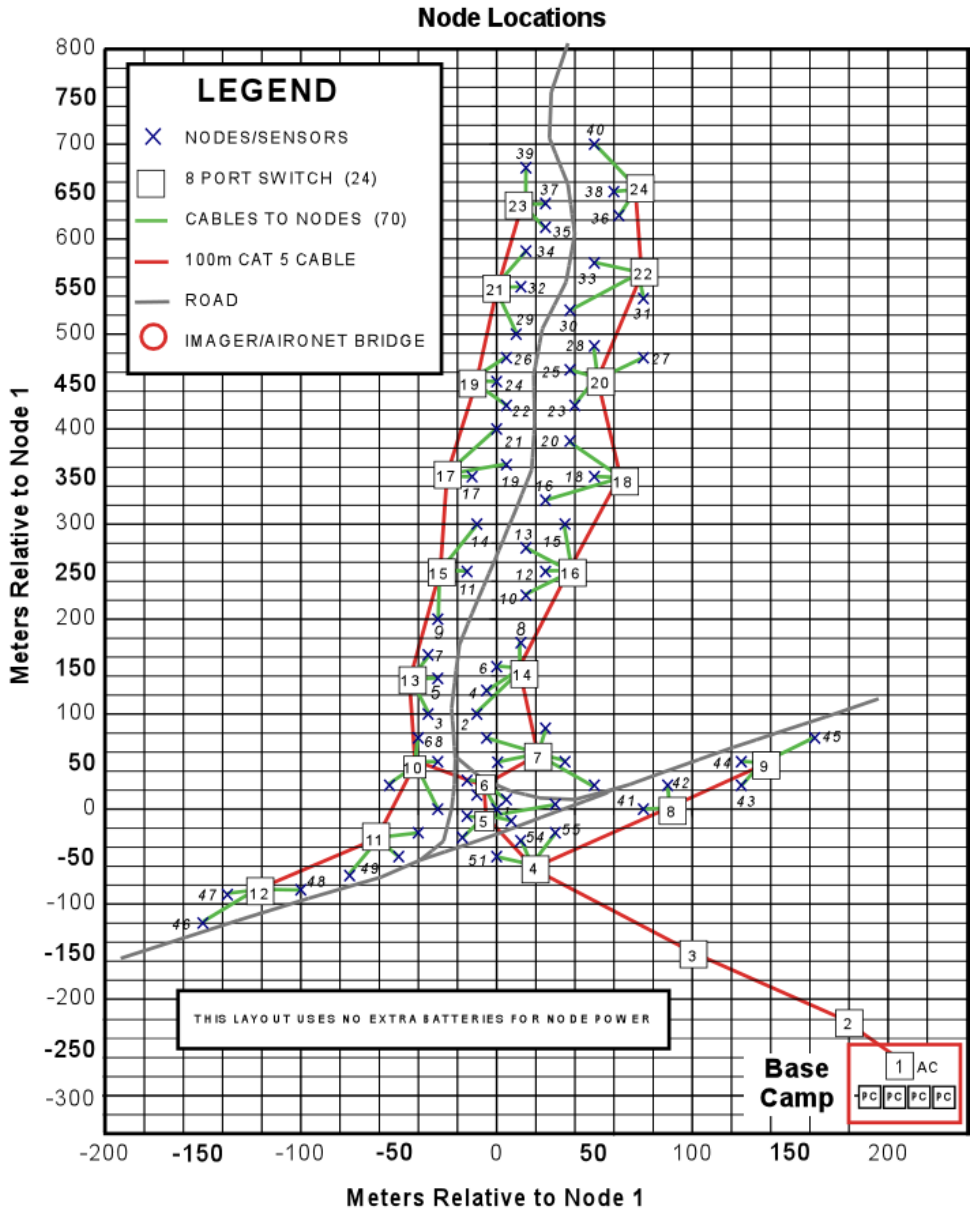
Baseline Experiment: Functional Architecture



November Baseline Participants



The Network


Node Locations
LEGEND
NODES/SENSORS
8 PORT SWITCH (24)
CABLES TO NODES (70)
100m CAT 5 CABLE
ROAD
IMAGER/AIRONET BRIDGE
THIS LAYOUT USES NO EXTRA BATTERIES FOR NODE POWER
Base Camp
1 AC
PC PC PC PC
Meters Relative to Node 1
Meters Relative to Node 1
800
750
700
650
600
550
500
450
400
350
300
250
200
150
100
50
0
-50
-100
-150
-200
-250
-300
-200
-150
-100
-50
0
50
100
150
200

Baseline Experiment: Accomplishments

- **Installed and operated in the field 70 dual-networked sensor nodes.**
  - **One of the largest wireless distributed sensor networks ever**
- **Collected 30GB time series and event data**
  - **significant expansion to SensIT's database of signature data (including HMMWV, 5-ton Truck, Dragonwagon, AAV, LAV, M1A1 Tank, and Soviet BMP-2)**
  - **provides data for further testing (and for playback)**
- **Established and tested procedures for**
  - **bringing node network up and down**
  - **initializing and operating an integrated set of processes on the Sensoria nodes**
- **Automatically triggered an imager node using track projections calculated from BAE event detections**
  - **Detections and trigger commands transmitted via diffusion**

SITEX02

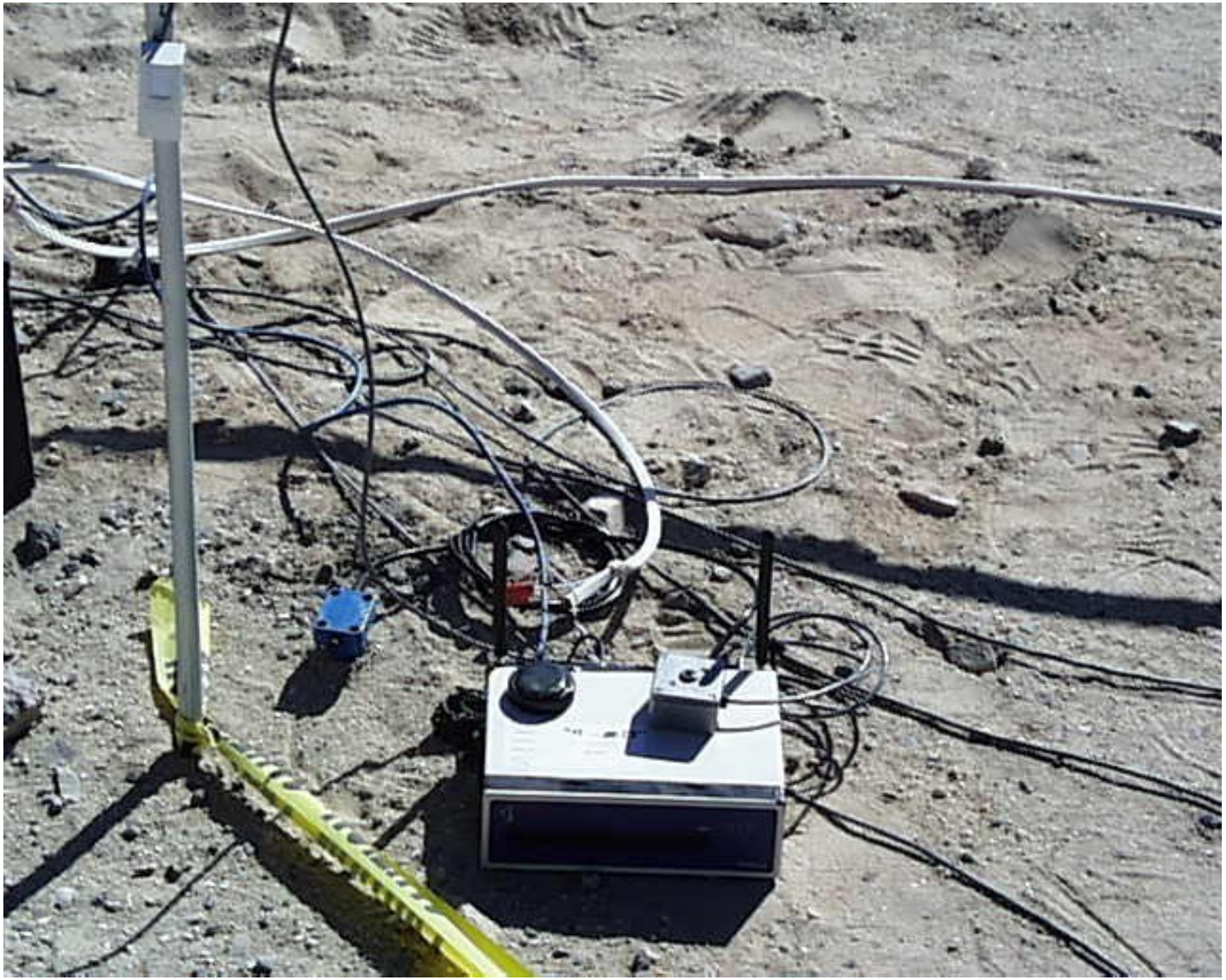
SITEX02 - MCAGCC, 29 Palms, CA

- 3 long weeks in the desert
- 53 Participants
- 16 Organizations
- 70 Nodes
- 1 Demo day
- 1 km of CAT 5 & Romex Cable
- >quart of suntan lotion
- Gallons of water
- 30 Marine Batteries
- 1 Destroyed Base Camp
- 1 Dead Car Battery
- Many late nights/early mornings

And they'd do it again...

Base Camp (aka Ford Excursion, post-Win

Dear to our hearts…and index fingers



The A-Team Battery Crew

Camp Bravo

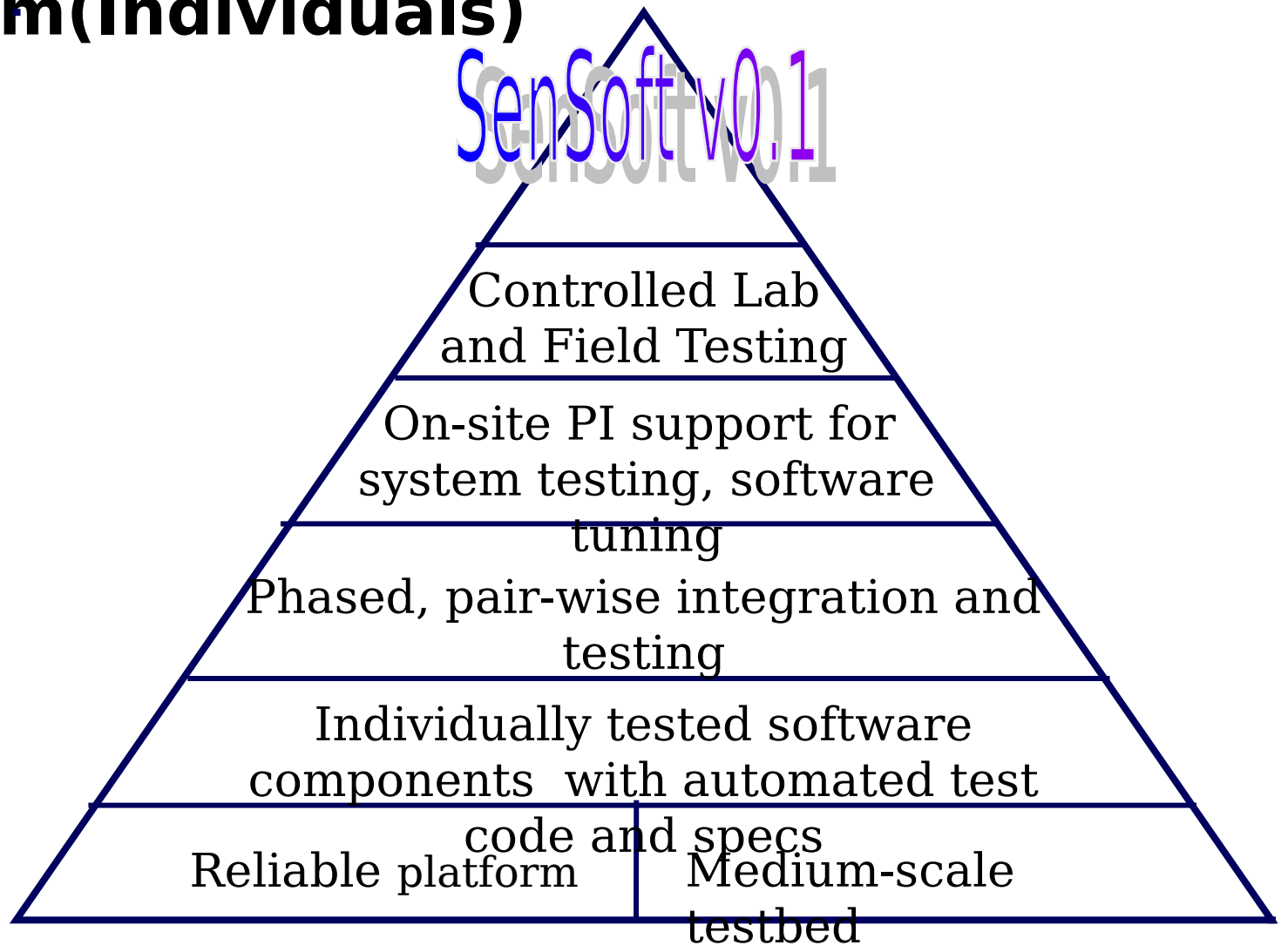
Room for Improvement

- **Network assembly**
  - **Radio cluster definitions manually constructed in SITEX02**
- **Network latency**
  - **Still needs to be quantified**
  - **Network behavior under significant loading not yet measured**
- **Platform improvements**
  - **Time synchronization (inter- and intra-node) and GPS location**
  - **Nodes required manual intervention when problems appeared**
- **Data access**
  - **Display of nodes, events, tracks not demonstrated as planned**
- **Query**
  - **Fixed query used in SITEX02 - “show all nodes, events and tracks”**

Going Forward

- Robust baseline system software suite
- Network status monitoring (self-test)
- Time synchronization & node localization (on-board GPS)
- Fault tolerant CSIP algorithms
- Network self-assembly
- Querying and tasking
  - easy to use fixed and mobile user interfaces
- Services-based architecture
- Software Interfaces
- Testing - integration and performance

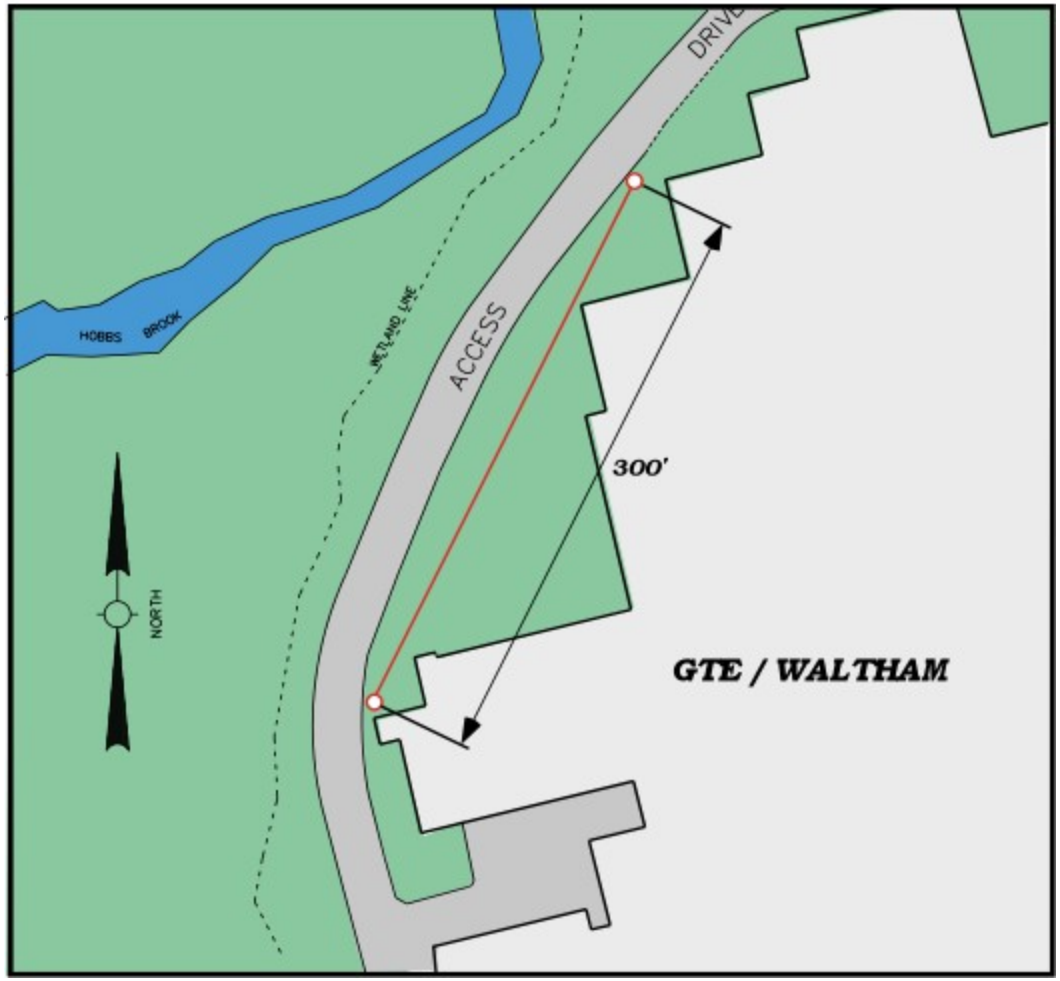
Integration by Parts: Whole > Sum(Individuals)


SenSoft v0.1
Controlled Lab and Field Testing
On-site PI support for system testing, software tuning
Phased, pair-wise integration and testing
Individually tested software components with automated test code and specs
Reliable platform
Medium-scale testbed

Planned 20 Node Testbed Site

emi-permanent medium-scale network for API development, debugging

re integration and testing using real targets (non-tactical) or hybrid sim



- **Road Frontage: ~ 300 feet**
- **Access to both sides of road**
- **AC Power Available**
- **Adjacent office space**
- **Low density traffic volume**
- **Video Surveillance**
- **TCP/IP Accessible**

Testbed - Looking North

Testbed - Looking South

Testbed - Adjacent Office & Open Area

SITEX02 Lessons Learned

- There's no substitute for being there!
- A large-scale network is essential for testing SensIT concepts , but imposes a significant logistical burden (manual reboot, battery replacement, etc.).
- Well-defined, open interfaces between functionalities are faster and easier to debug.
- Diagnostic tools can ease the test process: status checks, tracing, debugging, ...
  - Self-diagnostic nodes and radios, and nodes that can reboot without manual intervention would ease the logistical burden
- Reliable hardware is required for software testing. Similarly, debugged software is required before software integration
  - how to test? how to improve?
- Record everything!